SWEN90016

Software Processes & Project Management

Harry Drakos
Department of Computing and Information Systems
The University of Melbourne
harry.drakos@unimelb.edu.au

2019 – Semester 1

Lecture 2

L1 – Quiz

1. Feed back from previous semester.
2. Not marked.
3. Exam multiple choice questions will be selected from these.

Text **HARRYDRAKOS482** to **+61 427 541 357** once to join,

Lecture 1 – Recap

- ✓ Understand Assignments and our expectations
- ✓ Understand key elements of a Project and why organisations use them
- ✓ Understand the foundational components of Project Management
- ✓ Understand key skills, responsibilities & activities of a Project Manager
- ✓ Understand key elements of how to manage Projects
- ✓ Exposure to some Project Management Methodologies

Lecture 1 – Recap

- ✓ Explore key drivers in why projects fail / succeed
- ✓ Understand how organisations select the best / right projects
- ✓ Understand the Project Initialization process, Business Case structure and why organisations use them
- ✓ Explore various Investment techniques and financial models
- × Understand responsibilities associated with building a Business Case and the accountable group / individual
- × Understand what a Project Charter is and how it is used

L1.10 - Business Case. Who's is responsible for what?

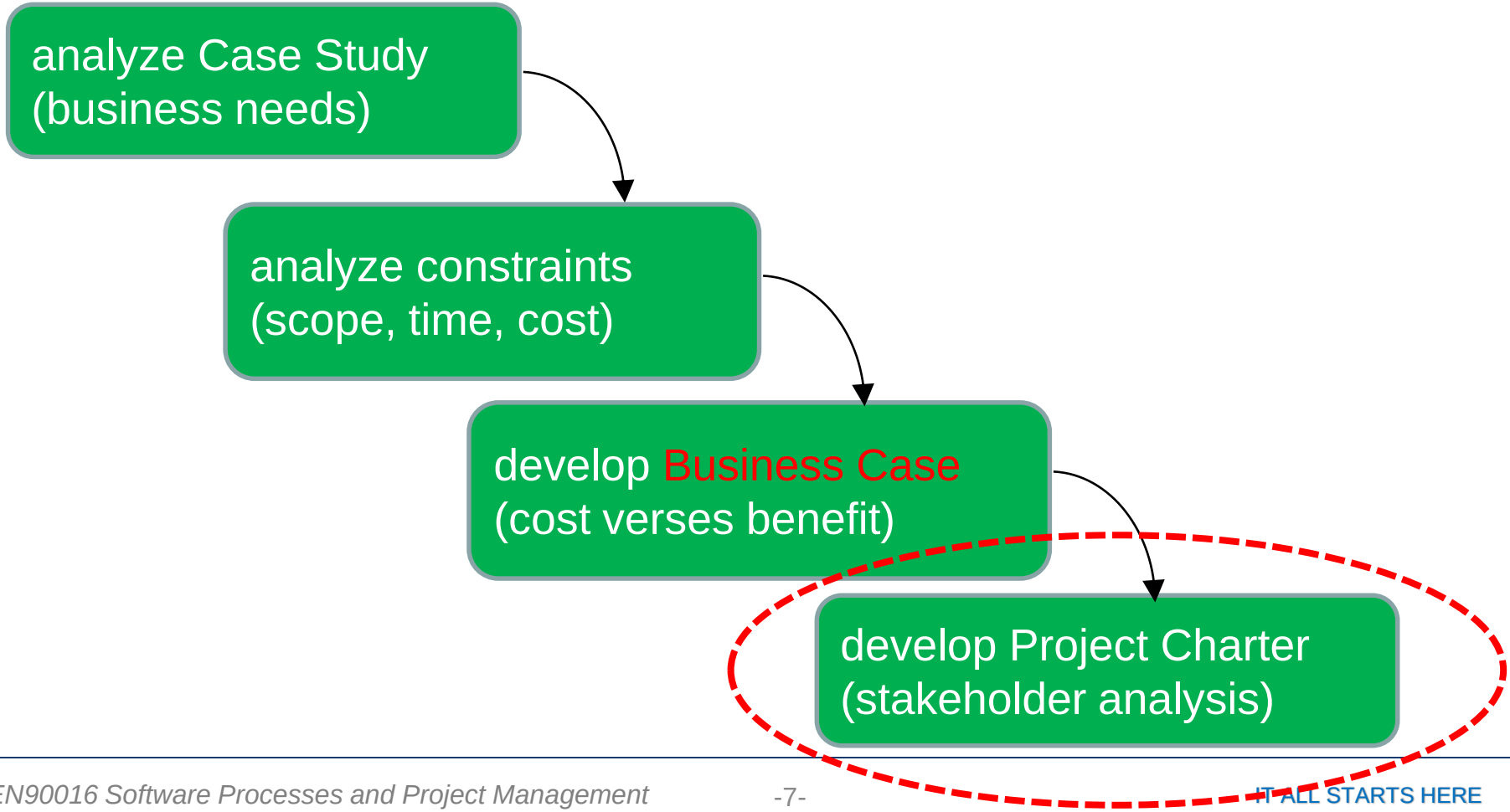
Role	Responsibilities
Corporate	1. Provides Mandate / The go ahead. 2. Holds Senior Users accountable for benefits realisation. 3. Responsible for conducting post projects benefits validation.
Executive / Sponsor	1. Owns the Business Case. 2. Responsible for reviewing the benefits throughout the project.
Senior Users	1. Responsible for accepting the benefits and delivering them. 2. Responsible for ensuring the delivered products are to the appropriate quality standard. 3. Provides on-going actual V forecasted benefit realisation.
Project Manager	1. Prepares the Business Case. 2. Conducts Risk assessment and impact analysis. 3. Assess and updates the Business Case at each defined stage.
Project Assurance / QA	1. Assists in developing the Business Case. 2. Ensure value for money and risks are continuously managed. 3. Monitors change to the Business Case and validates it.
Project Support	1. Responsible for capturing data and preparing management reports. 2. Key support point for all project stakeholders – schedules, cost analysis, minutes, actions, supplier liaison etc.

L1 – Intended Learning Objectives

6. ~~Explore the key drivers of why projects fail / succeed.~~
7. ~~Understand how organisations select the best / right projects (Project Screening).~~
8. ~~Understand the Project Initialization process, Business Case structure and why organisations use them.~~
9. ~~Explore various investment techniques and financial models.~~
10. ~~Understand responsibilities associated with building a Business Case and the accountable group / individual.~~
11. Understand what a Project Charter is and how it is used.

L1.11 – Starting up a Project – Project Charter

The first *process: initialization*



Lecture 1 - Simple Project Charter

Project Name

Target Date: [Date]

Project Description

Write out the project description here. Write out the project description here. Write out the project description here. Write out the project description here. Write out the project description here. Write out the project description here.

Costs

Item	Quantity	Rate	Total
Resources			
Equipment			
Budget			
Total			

Gains

Item	Quantity	Rate	Total
Cost Savings			
Time Savings			
Revenue Gain			
Net Total			

Project Team

- Person 1 – Project Manager
- Person 2 – Team Lead
- Person 3 – Analyst
- Person 4 – Developer
- Person 5 – Quality
- Person 6 – Trainer
- Person 7 – Other
- Person 8 – Other
- Person 9 – Other
- Person 10 – Other

Milestone 1
[Date]
[Description of what will be accomplished on this milestone]

Milestone 2
[Date]
[Description of what will be accomplished on this milestone]

Milestone 3
[Date]
[Description of what will be accomplished on this milestone]

L1 – Quiz

1. Feed back from previous semester.
2. Not marked.
3. Exam multiple choice questions will be selected from these.

Text **HARRYDRAKOS482** to **+61 427 541 357** once to join,

Key project characteristics include:

Introduce change

They are temporary - they have a start and end date

Cross functional

Vary in size

All of the above

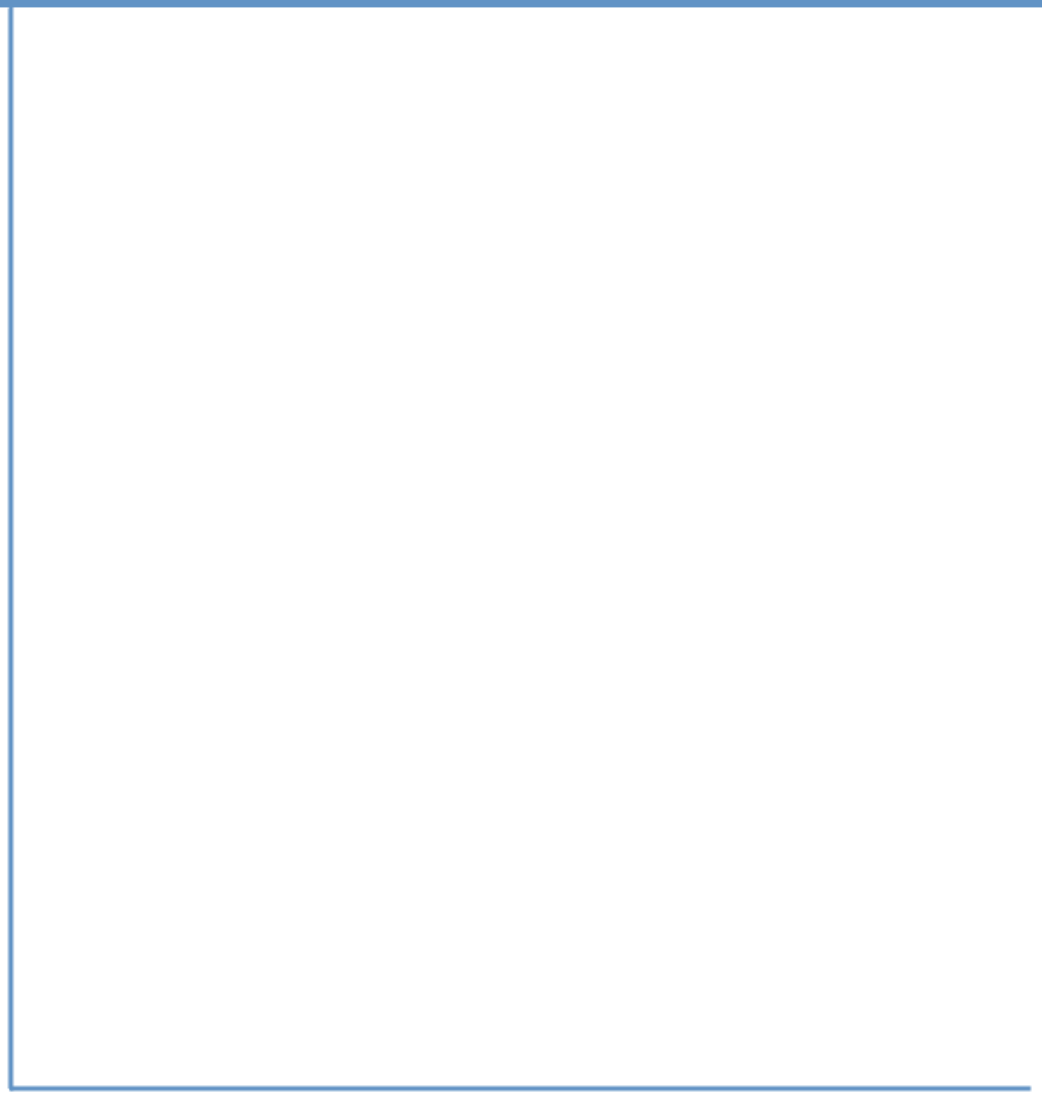
The value of Project Management lies in:

Organising and structuring scare resources and managing risk

Not needing to worry about costs or timeframes

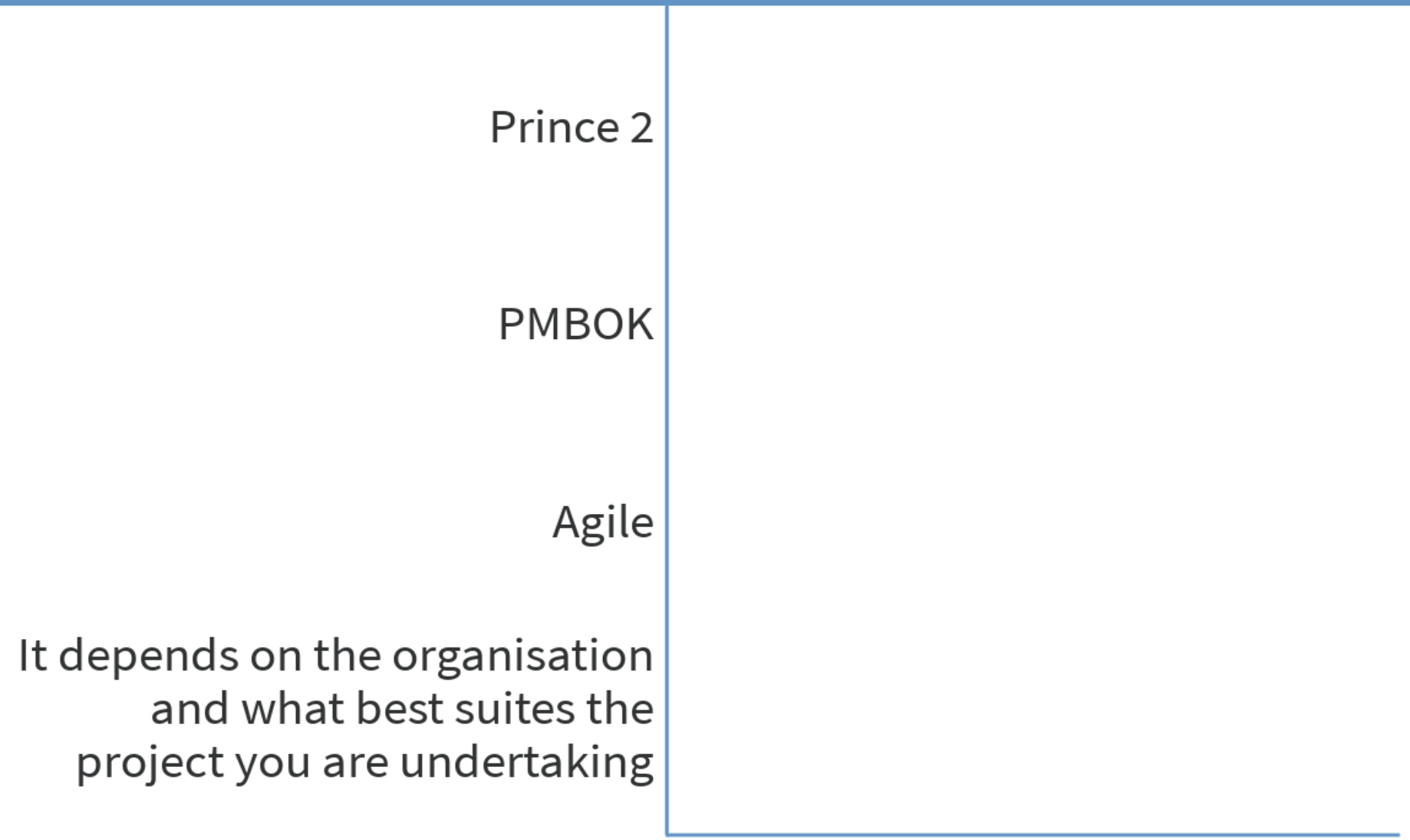
They are totally unstructured

Anyone in an organisation can start or stop one

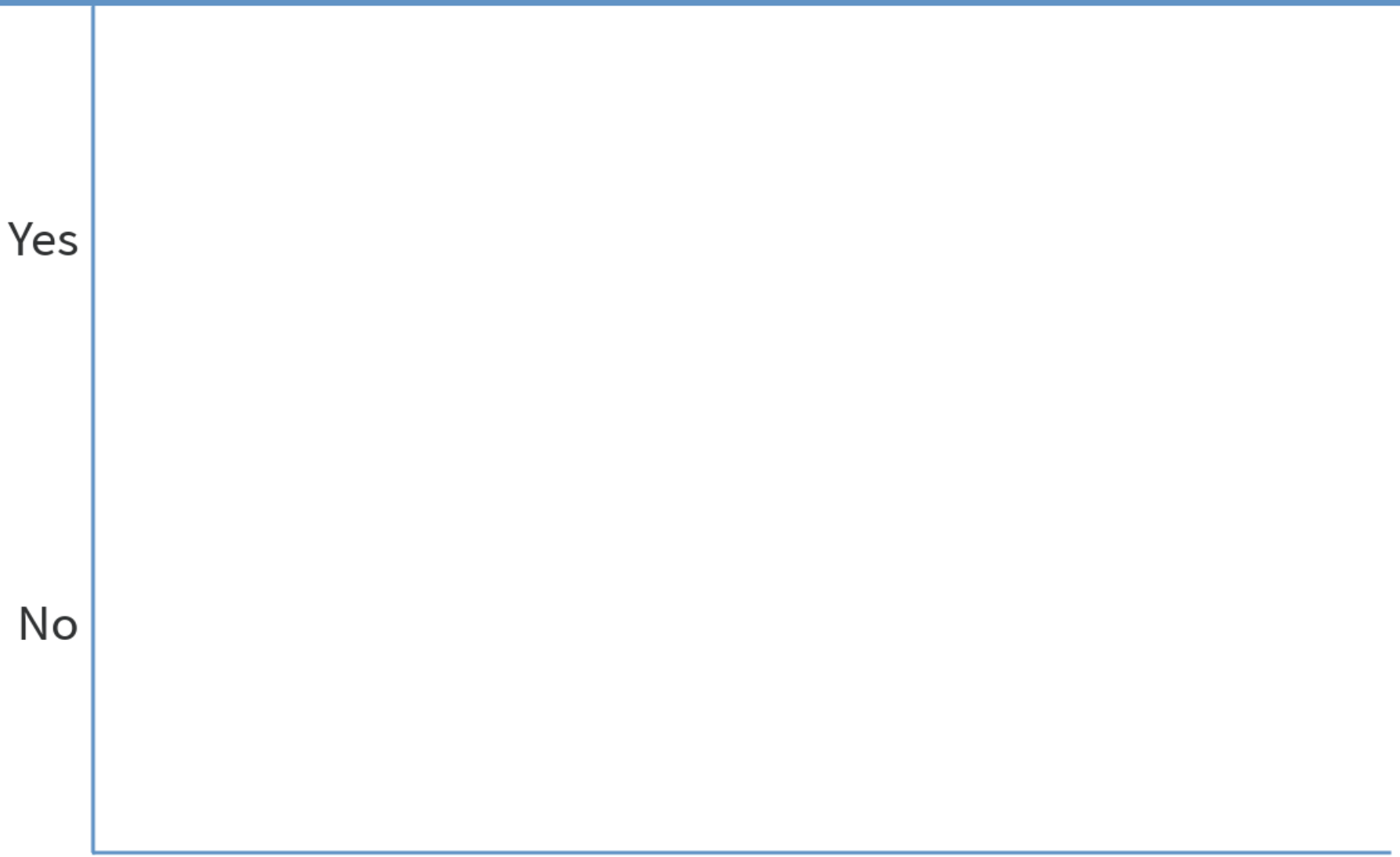


The #1 factor (according to Standish Group Research) that contributes to Project Failure is:

The best methodology / approach to use for projects is:



A Project Charter should be created for every project.



When is Assignment 1 due?

Week 8 – I have plenty of time and don't need to start any time soon	**A**
Week 2 (this week) – I can't believe these lecturers	**B**
Week 5 – I have a bit of time but I better start this week so I don't fall behind	**C**
Are you serious, we have 2 assignments	**D**

Semester Structure

Week #	Lecture Date	Lecture Law G15 – Thursday 9.00am to 11.00am	Assignment
1	07/03/19	Subject Introduction, Introduction to Projects and Project Management,	
2	**14/03/19**	**Project Management Plan & SDLC's**	**Assignment 1 Spec available on LMS 15/3**
3	21/03/19	SDLC - Agile Scrum – continued Individuals, Motivation and Teams	
4	28/03/19	Stakeholder Management Communication Management	Assignment 2 available & Groups created during the workshops / tutorials – attendance mandatory
5	04/04/19	Project Planning and Scheduling Assignment 1 & 2 open forum / discussion	Assignment 1 (Individual) due Fri 5/4 @ 11.59 pm
6	11/04/19	Cost Estimation	
7	18/04/19	Risk Management	
	25/04/19	***Non Teaching Week – Mid semester break***	Assignment 2 (Part 1) due Wed 24/4 @ 11.59 pm
8	02/05/19	Quality Management	
9	09/05/19	Configuration Management (including guest lecture)	Assignment 2 (Part 2) due Sat 11/5 @ 11.59 pm
10	16/05/19	Ethics, Outsourcing & Procurement	Assignment 2 (Part 3) due Sat 18/5 @ 11.59 pm
11	23/05/19	Guest Lecture	Assignment 2 (Final) due Sat 25/5 @ 11.59 pm
12	30/05/19	Subject Revision and Exam Prep	Assignment 2 Project Demonstration during tutorials

L2 - Intended Learning Objectives

1. Understand what a Process is and its relevance to Project Management.
2. Understand what a Project Management Plan (PMP) is and when it should be used.
3. Understand the components of a Project Management Plan.
4. Understand what a Software Development Lifecycle (SDLC) is and the advantages / disadvantages of various models.

L2 - Intended Learning Objectives

5. Understand what Agile is and its origins.
6. Understand the Agile framework.
7. Understand Scrum – Roles, Ceremonies and Artefacts.
8. Understand advantages / disadvantages of Agile.
9. Understand key questions that will help select which approach to use and some examples.

L2 - Intended Learning Objectives

1. Understand what a Process is and its relevance to Project Management.
2. Understand what a Project Management Plan (PMP) is and when it should be used.
3. Understand the components of a Project Management Plan.
4. Understand what a Software Development Lifecycle (SDLC) is and the advantages / disadvantages of various models.

L2.1 – What is a Process

A. A set of actions or activities that are randomly completed to achieve any outcome.

B. A series of steps to achieve a particular end.

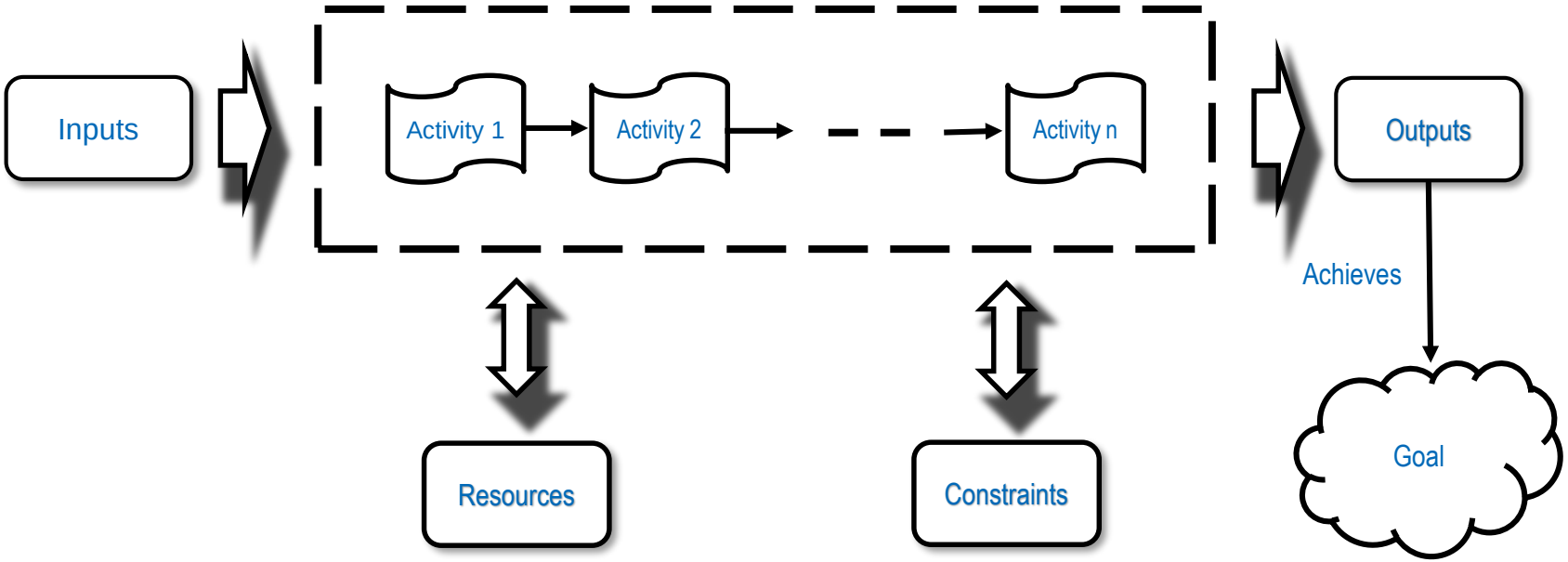
C. A specific activity that is partially completed.

D. All the Above

Definition*: "A process is a series of progressive and interdependent steps by which an end is attained."*

L2.1 – What is a Process?

A process is a series of progressive and interdependent steps by which an end is attained.



L2.1 – What does a Process have to do with Project Management and Software Engineering?

1. Project Management is a process as it defines a series of tasks (Planning, Executing and Controlling Projects) to deliver a specific / agreed outcome.

2. System Development Lifecycle (SDLC) is a term used in Software Engineering. It describes a process for planning, creating, testing, and deploying an information system. SDLC can be composed of hardware only, software only, or a combination of both.

L2 - Intended Learning Objectives

1. ~~Understand what a Process is and its relevance to Project Management.~~
2. Understand what a Project Management Plan (PMP) is and when it should be used.
3. Understand the components of a Project Management Plan.
4. Understand what a Software Development Lifecycle (SDLC) is and the advantages / disadvantages of various models.

L2.2 – Project Management is a Process

Lecture 1

Step 1:
Screen candidate projects & seek approval

Lecture 1

Step 2:

Business Case:
document organizational buy-in

Step 5:
Deliver Benefits
Final Project Report
Lessons Learnt

Lecture 2

Step 3:
Manage Project – Project Management Plan

Step 4:
Build & Deploy - via SDLC

Source: www.pmi.org/pmbok-guide-standards

L2.2 – Project Management Plan (Formal)

Almost every organisations will have it's own *"version"* of a Project Management Plan (PMP), however the reasons they have and use them are the same.

A PMP is a formal approved document that defines how the project is executed, monitored and controlled. It may be a summary or a detailed document.

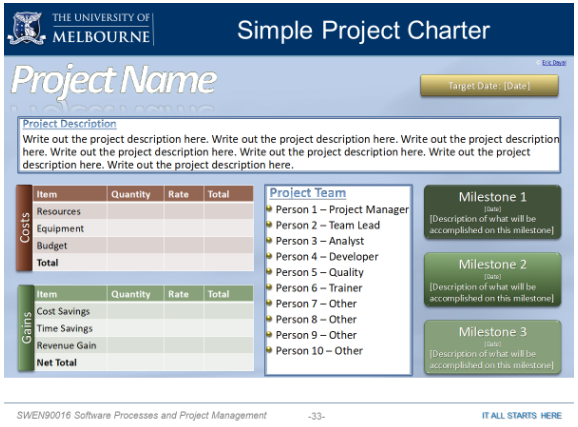
It is a document that is owned, controlled and populated by the Project Manager and is used throughout the project.

A good PMP provides the required level of detail across key project components and is the one source of truth for all parties involved across the project.

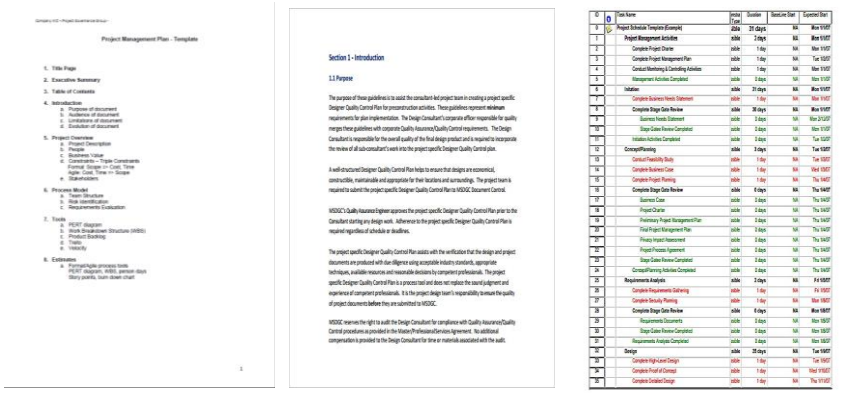
L2.2 – Project Charter V Project Management Plan

A Project Charter is a summary project proposal to secure approval for the project goals and terms (useful as part of Business Case).

A PMP is an approved document showing how to achieve the approved project goals / benefits and provides the details on how to execute and manage the project (used as part of mobilisation and on-going management of the project).



Primary Use: Summary (few pages) of key information used to communicate, engage, gain buy-in and obtain approvals.



Primary Use: Detailed document used to establish and manage the project. Defines all key items the project needs to consider.

L2 - Intended Learning Objectives

1. ~~Understand what a Process is and its relevance to Project Management.~~
2. ~~Understand what a Project Management Plan (PMP) is and when it should be used.~~
3. Understand the components of a Project Management Plan.
4. Understand what a Software Development Lifecycle (SDLC) is and the advantages / disadvantages of various models.

L2.3 – Project Management Plan (Formal)

A typical PMP consists of all / or most of the following categories.

- *Project Information*
  - Executive Summary
  - Financial Authority to proceed
  - Key Stakeholders
  - Scope
  - Delivery approach / SDLC - Waterfall or Agile
  - Resources / People
  - Key Milestones
  - Project Budget
  - Business Value (Financial & Non-Financial Benefits)
  - Lessons learned applied to this project
  - Constraints

L2.3 – Project Management Plan (Formal)

A typical PMP consists of all / or most of the following categories.

- *Project Governance*
  - Roles and Responsibilities
  - Mandatory Project Planning / Key Additional Activities
    - Schedule
    - Risk Management
    - Cost Estimation
    - Quality Assurance
    - Configuration Management (Change Management)

The PMP is a large multi-page document that takes time to prepare, review and complete. Multiple people (subject experts) are involved and prepare the specific details. The Project Manager coordinates all items and has ultimate accountability for the quality and final outcome.

L2 - Intended Learning Objectives

1. ~~Understand what a Process is and its relevance to Project Management.~~
2. ~~Understand what a Project Management Plan (PMP) is and when it should be used.~~
3. ~~Understand the components of a Project Management Plan.~~
4. Understand what a Software Development Lifecycle (SDLC) is and the advantages / disadvantages of various models.

L2.4 – SDLC is a Process

Lecture 1

Step 1:
Screening approving project to proceed

Lecture 1

Step 2:
Business Case to ensure organisational buy-in

Step 3:
Manage Project – Project Management Plan

Lecture 2

Step 4:
Build & Deploy - via SDLC

Lecture 2

Step 5:
Deliver Benefits
Final Project Report
Lessons Learnt

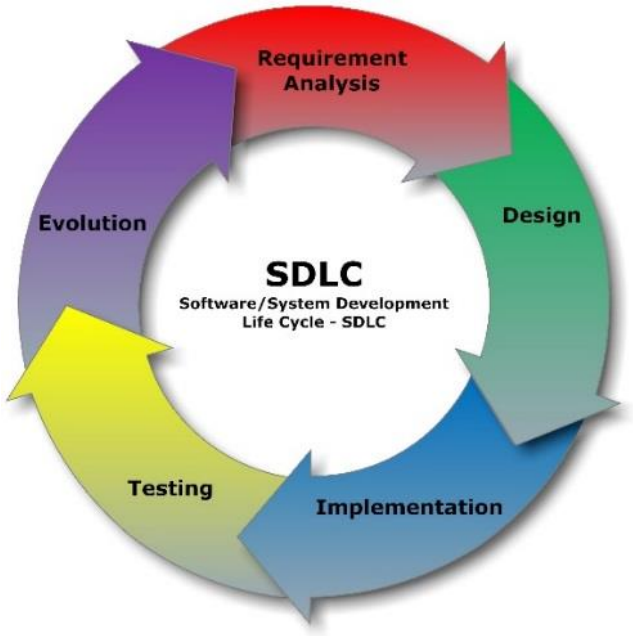
Source: www.pmi.org/pmbok-guide-standards

L2.4 – Software Development Life Cycle (SDLC)

The systems development life cycle (SDLC), also referred to as the application development life-cycle, is a term used in systems engineering, information systems and software engineering to describe a ***process*** for planning, creating, testing and deploying an information system.

Activities in SDLC:

- Requirements gathering
- Systems / Architectural Design
- Implementation / coding
- Integration
- Testing
- Delivery and Release - Deployment
- Maintenance



L2.4 – SDLCs

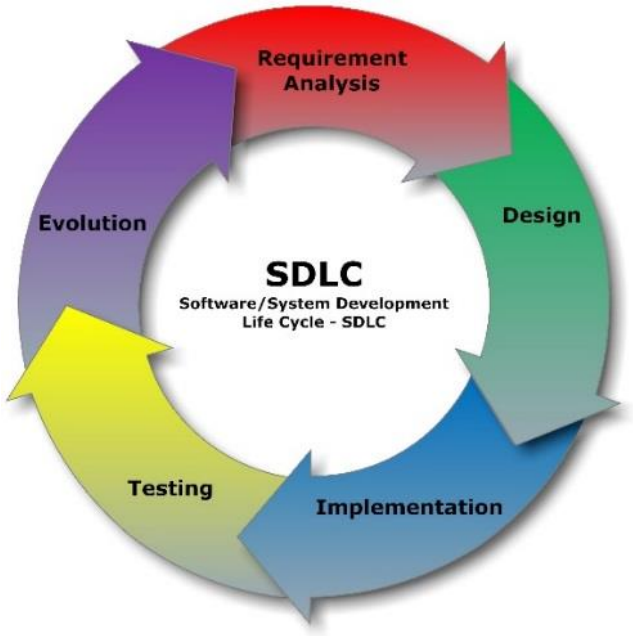
There are many SDLCs around with organisations typically favouring a blend of Formal and Agile approaches.

Formal Processes

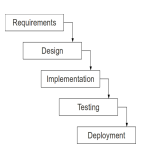
- Waterfall
- Incremental
- V-Model

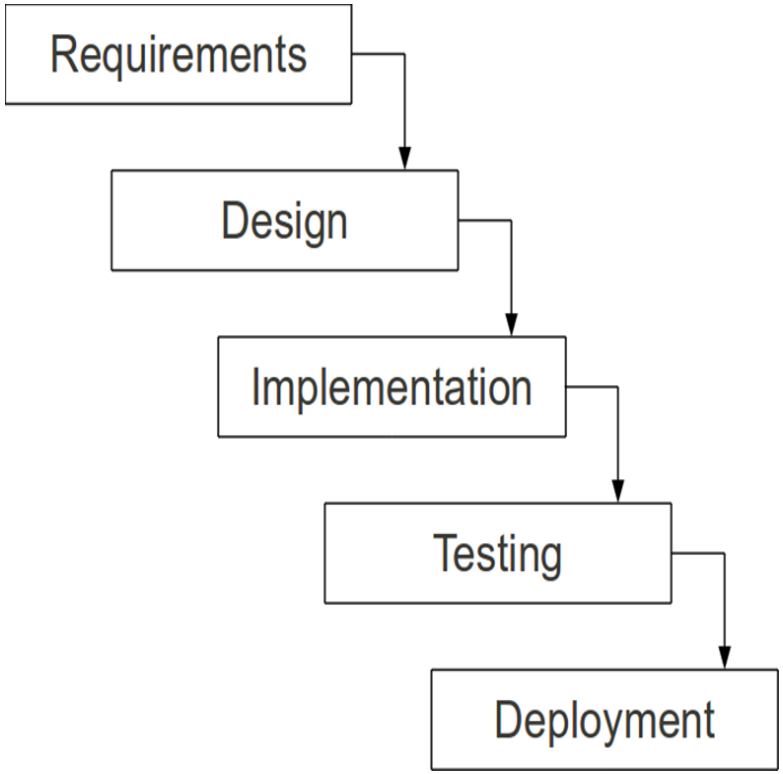
Agile Processes

- Extreme Programming
- Scrum
- Kanban



L2.4 – SDLC - Waterfall





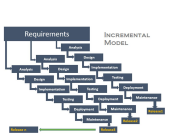
Advantages

- Simple and easy to understand and use
- Easy to manage due to the rigidity of the model
- Phases are processed and completed one at a time
- Documentation available at the end of each phase
- Works well for projects where requirements are very well understood and remain stable

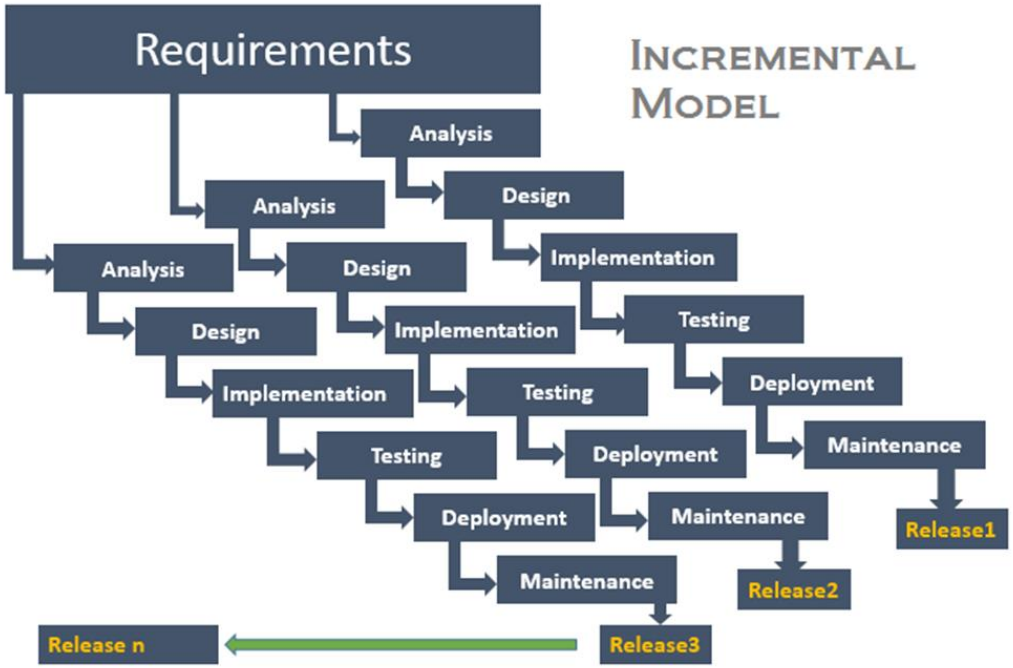
Disadvantages

- Difficult to accommodate change after the process in underway
- One phase must be completed before moving on to the next
- Unclear requirements lead to confusion
- Clients approval is in the final stage
- Difficult to integrate risk management due to uncertainty

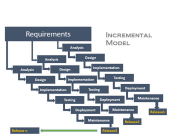
L2.4 – SDLC – Incremental Model



In incremental model the ***whole requirement*** is divided into various releases. Multiple cycles take place, making the life cycle a ***multi-waterfall*** cycle. Cycles are divided up into smaller, more easily managed modules.



L2.4 – SDLC – Incremental Model



Advantages

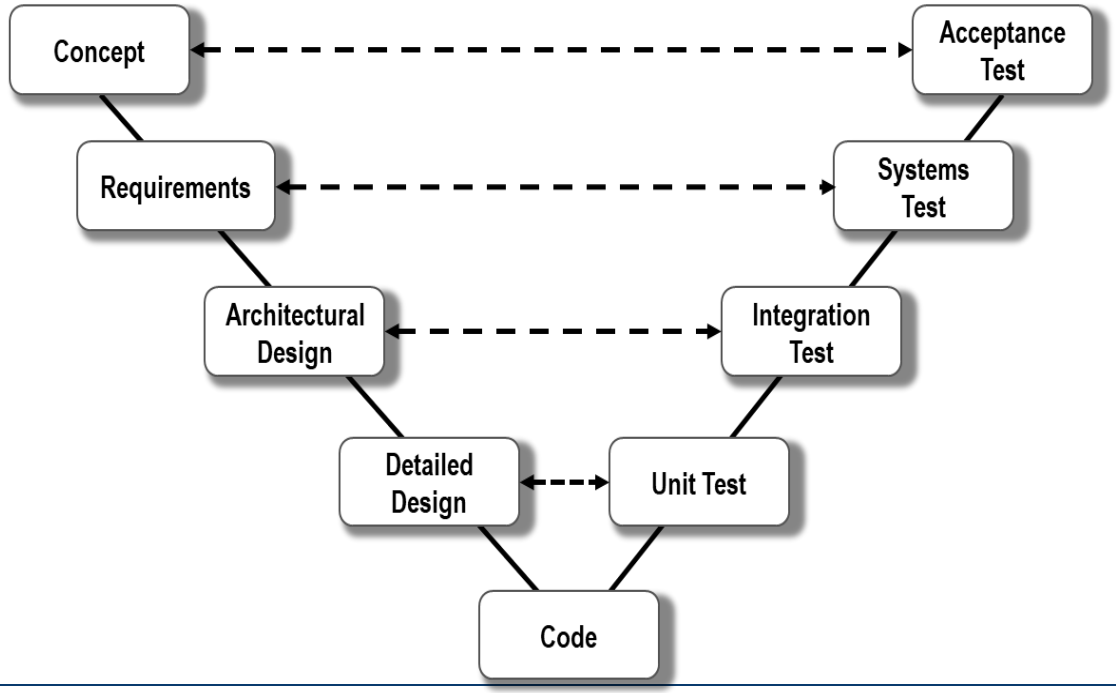
- Each release delivers an operational product
- Less costly to change the scope/requirements
- Customers can respond to each build
- Initial product delivery is faster
- Customers get important functionality early
- Easier to test and debug during smaller iterations

Disadvantages

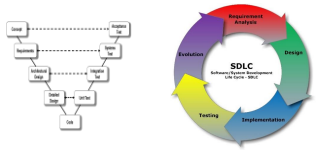
- More resources may be required
- More management attention is required
- Defining / partitioning the increments is difficult and often not clear
- Each phase of an iteration is rigid with no overlaps
- Problems may occur at the time of final integration

L2.4 – SDLC – V-Model

The V-Model is an extension of the waterfall model and is based on the association of a testing phase for each corresponding development stage. This means that for every single phase in the development cycle, there is a directly associated testing phase. This is a highly-disciplined model and the next phase starts only after completion of the previous phase



L2.4 – SDLC – V-Model



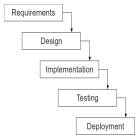
Advantages

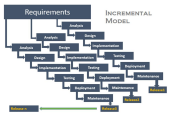
- Simple and easy to understand and use
- Each phase has specific deliverables and well defined objectives and goals
- High chance of success over waterfall model due to the development of test plans early on during life cycle
- Works well for small projects when requirements are easily understood

Disadvantages

- Very rigid process like the waterfall model
- Little flexibility and adjusting scope is difficult and expensive
- Software developed during implementation phase, no early prototypes
- No clear path for problems found during testing
- Changes in later stage cause test documentation across all stages to be changed

L2.4 – SDLC – Formal Models






Characteristics where "Formal" Models make sense to use

- Projects where the customer has a very clear view of what they want
- Projects that will require little or no change to requirements
- Software requirements are clearly defined and documented
- Software development technologies and tools are well-known
- Large scale applications and systems developments

Lecture Break

BREAK

Please return promptly as the

Lecture will re-start in *5 mins*

L2 - Intended Learning Objectives

5. Understand what Agile is and its origins.
6. Understand the Agile framework.
7. Understand Scrum – Roles, Ceremonies and Artefacts.
8. Understand advantages / disadvantages of Agile.
9. Understand key questions that will help select which approach to use and some examples.

L2.5 – SDLCs

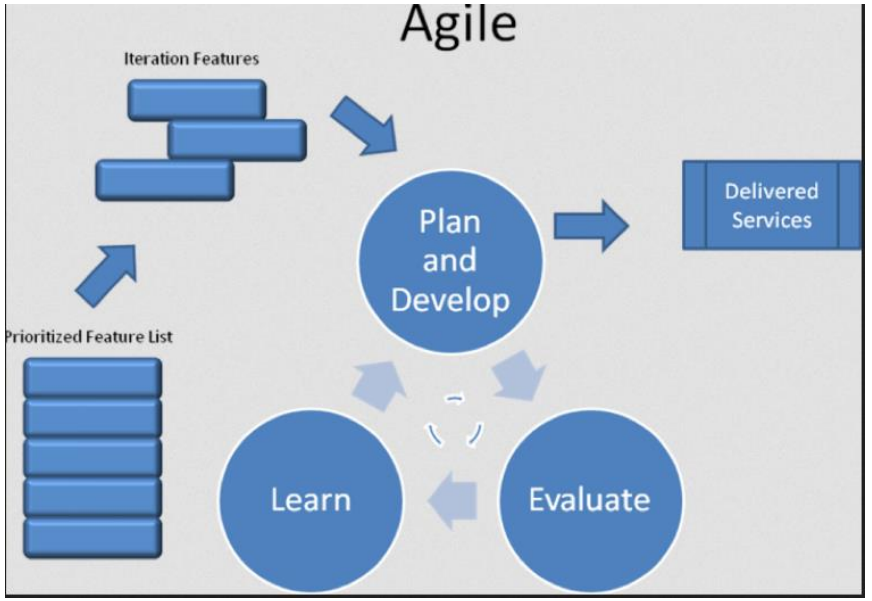
There are many SDLCs around with organisations typically favouring a blend of Formal and Agile approaches.

Formal Processes

- Waterfall
- Incremental
- V-Model

Agile Processes

- Kanban
- Extreme Programming
- Scrum



L2.5 – Why is Agile attractive

- We are in an ever changing global world with the pace of change increasing
- Customer needs and demands are exponentially increasing – products must continually be delivered
- Low Technology cost, ease of use and the global market place has increased competition and reduced entry barriers
- The war for talent is over – and we have lost! Cross functional teams help minimise the potential loss
- Long development cycles are like long lunches – a thing of the past
- Quality is no longer something we do / check later – it must be part of everything we do
- Cross functional groups are more fun!

L2.5 – What is Agile

- A set of methodologies based on iterative development where requirements and solutions evolve through collaboration between self-organising cross-functional teams
- A disciplined process that encourages frequent inspection and adaptation
- A leadership philosophy that encourages teamwork, self-organisation and accountability
- A set of engineering best practices intended to allow for rapid delivery of high-quality software
- A business approach that aligns development with customer needs and company goals

L2.5 – What is Agile

- In software development, we think about methodologies, activities, interactions, results, work products, artefacts and processes to organise the work.

The main tasks of software development remain the same regardless of methodology used, however with Agile, the flow of activities, how they are undertaken and who is involved is extremely different.

L2.5 – What is Agile - Origins

- Changes initiated in large US corporates in 1990's
- Software engineers frustrated with
  - long lead times before products were delivered
  - Decisions made early in the project couldn't be changed later
- 17 software engineer thought leaders met first in 2000 to discuss software engineering and different approaches
- Famous meeting in 2001 at the Snowbird ski lodge in Utah where they met to change the way the industry designed, engineered and deployed software
- Brought together by the Agile Manifesto

https://techbeacon.com/agility-beyond-history%E2%80%94-legacy%E2%80%94-agile-development

L2 - Intended Learning Objectives

5. ~~Understand what Agile is and its origins.~~
6. Understand the Agile framework.
7. Understand Scrum – Roles, Ceremonies and Artefacts.
8. Understand advantages / disadvantages of Agile.
9. Understand key questions that will help select which approach to use and some examples.

L2.6 – Agile Framework

Primary elements of the Agile framework include:

- Manifesto
- 12 Key Principles
- Kanban
- Scrum

L2.6 – Agile Framework - *Manifesto*

Agile Manifesto

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

Individuals and interactions	over	processes and tools
Working software	over	comprehensive documentation
Customer collaboration	over	contract negotiation
Responding to change	over	following a plan

That is, while there is value in the items on the ***right***, we value the items on the ***left*** more.

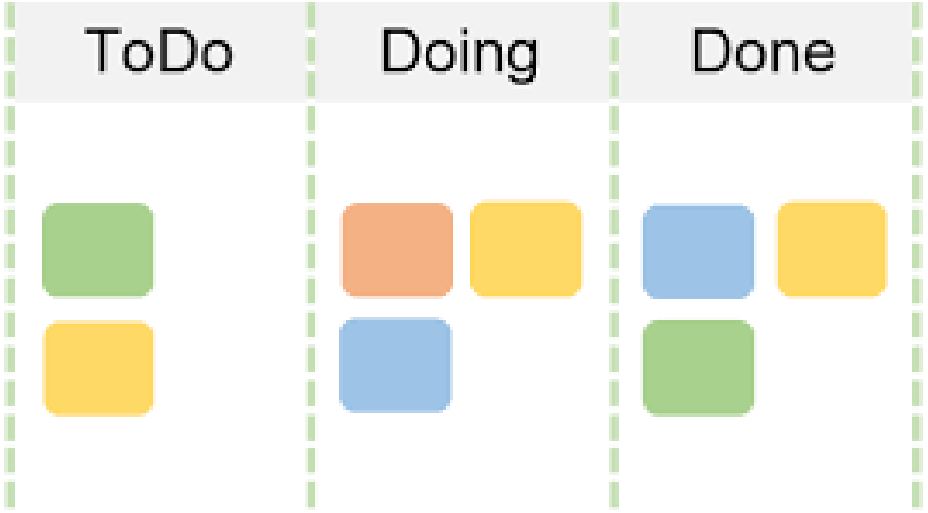
http://www.agilealliance.org

L2.6 – Agile Framework – *12 Key Principles*

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
2. Welcome changing requirements, even late in development. Agile harness change for the customer's competitive advantage.
3. Deliver working software frequently, from a couple of weeks to a couple of months, shorter timeframes is the preference.
4. Business people and developers must work together daily throughout the project.
5. Build projects around motivated individuals. Give them the environment and support they need and trust them.
6. The most efficient and effective method of conveying information to and within a development team is face-to-face
7. Working software is the primary measure of progress.
8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
9. Continuous attention to technical excellence and good design enhances agility.
10. Simplicity - the art of maximizing the amount of work not done - is essential.
11. The best architectures, requirements, and designs emerge from self-organising teams.
12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

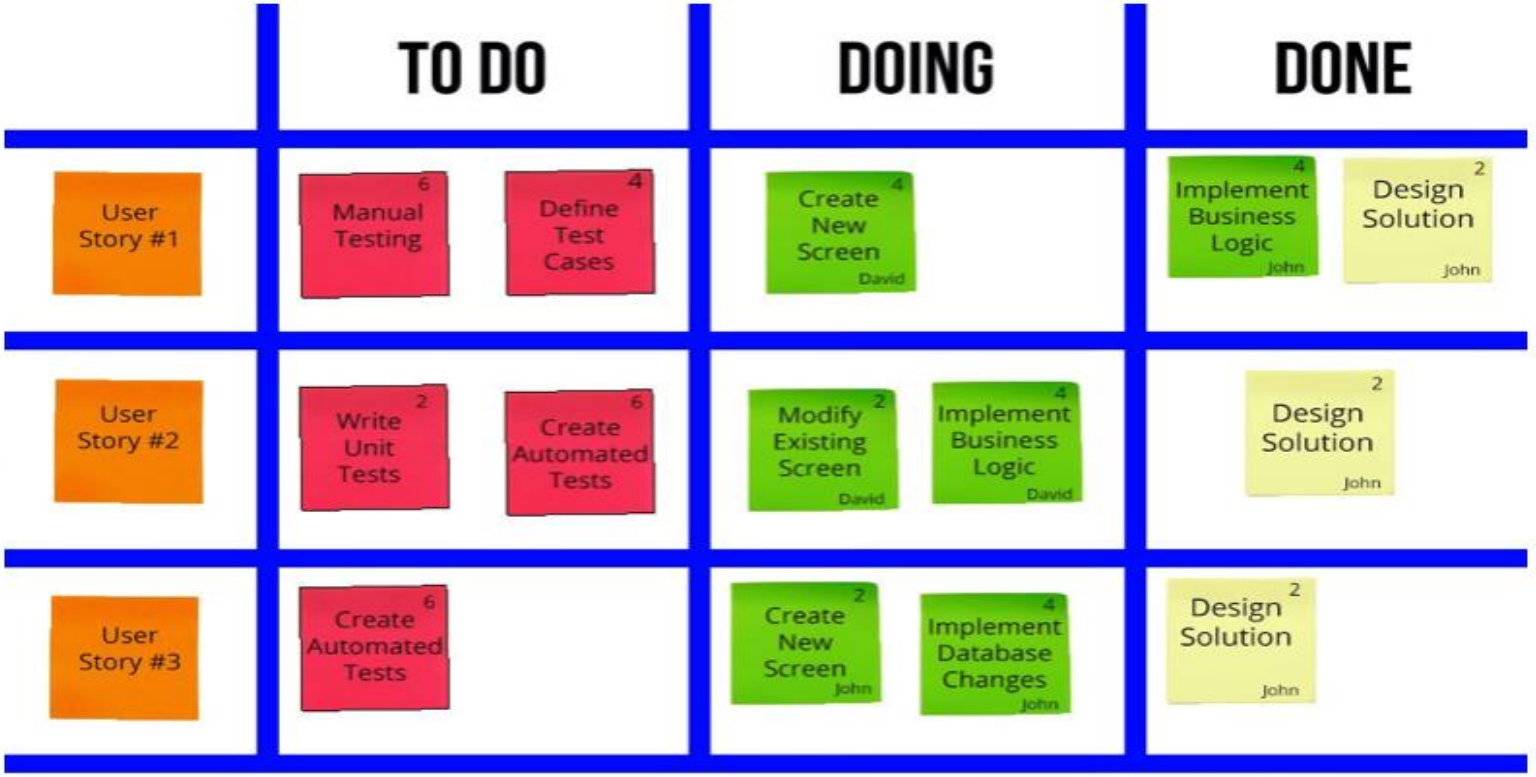
L2.6 – Agile Framework - *Kanban*

- ***Signboard / Billboard:*** Work items are visualised to provide participants a view of progress and process, from start to finish usually via a Kanban board



L2.6 – Agile Framework - *Kanban*

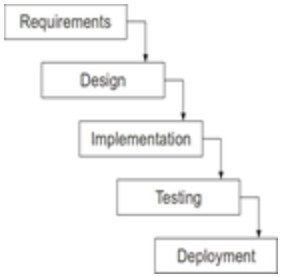
- Visual progress gives transparency/accountability for self-organizing teams called the ***SWIMLANE*** boards in other SWEN90016 resources



L2.6 – Agile Framework - *Scrum*

Scrum is an agile way to manage a project

> "The… 'relay race' approach to product development…may conflict with the goals of maximum speed and flexibility. Instead a holistic or 'rugby' approach—where a team tries to go the distance as a unit, passing the ball back and forth—may better serve today's competitive requirements."
>
> Hirotaka Takeuchi and Ikujiro Nonaka, "The New New Product Development Game", *Harvard Business Review*, January 1986.



L2 - Intended Learning Objectives

5. ~~Understand what Agile is and its origins.~~
6. ~~Understand the Agile framework.~~
7. Understand Scrum – Roles, Ceremonies and Artefacts.
8. Understand advantages / disadvantages of Agile.
9. Understand key questions that will help select which approach to use and some examples.

L2.7 – Scrum

Scrum in 100 words

- Scrum is an agile process that allows us to focus on delivering the highest business value in the shortest time.
- It allows us to rapidly and repeatedly inspect actual working software (every two to four weeks).
- The business sets the priorities. Teams self-organise to determine the best way to deliver the highest priority features.
- Every two to four weeks, you can see real working software and decide to release it as is or continue to enhance it for another sprint.

L2.7 – Scrum is used by many organisations

- Microsoft
- Yahoo
- Google
- Electronic Arts
- High Moon Studios
- Lockheed Martin
- Philips
- Siemens
- Nokia
- Capital One
- BBC
- Intuit
- Intuit
- Nielsen Media
- First American Real Estate
- BMC Software
- Ipswitch
- John Deere
- Lexis Nexis
- Sabre
- Salesforce.com
- Time Warner
- Turner Broadcasting
- Oce

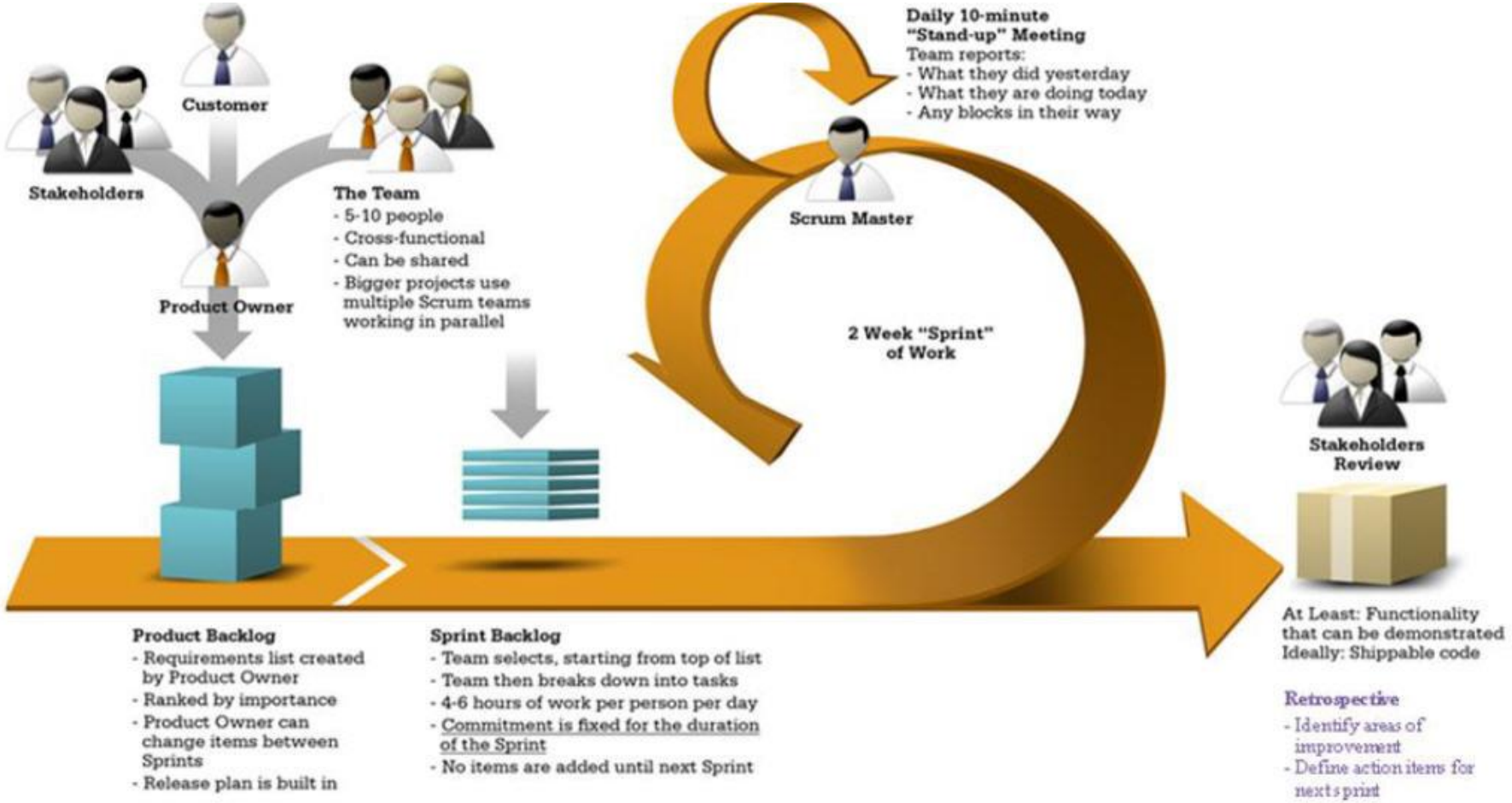
L2.7 – Scrum is used for all types of projects

- Commercial software
- In-house development
- Contract development
- Fixed-price projects
- Financial applications
- ISO 9001-certified applications
- Embedded systems
- 24x7 systems with 99.999% uptime requirements
- the Joint Strike Fighter
- Video game development
- FDA-approved, life-critical systems
- Satellite-control software
- Websites
- Handheld software
- Mobile phones
- Network switching applications
- ISV applications
- Some of the largest applications in use

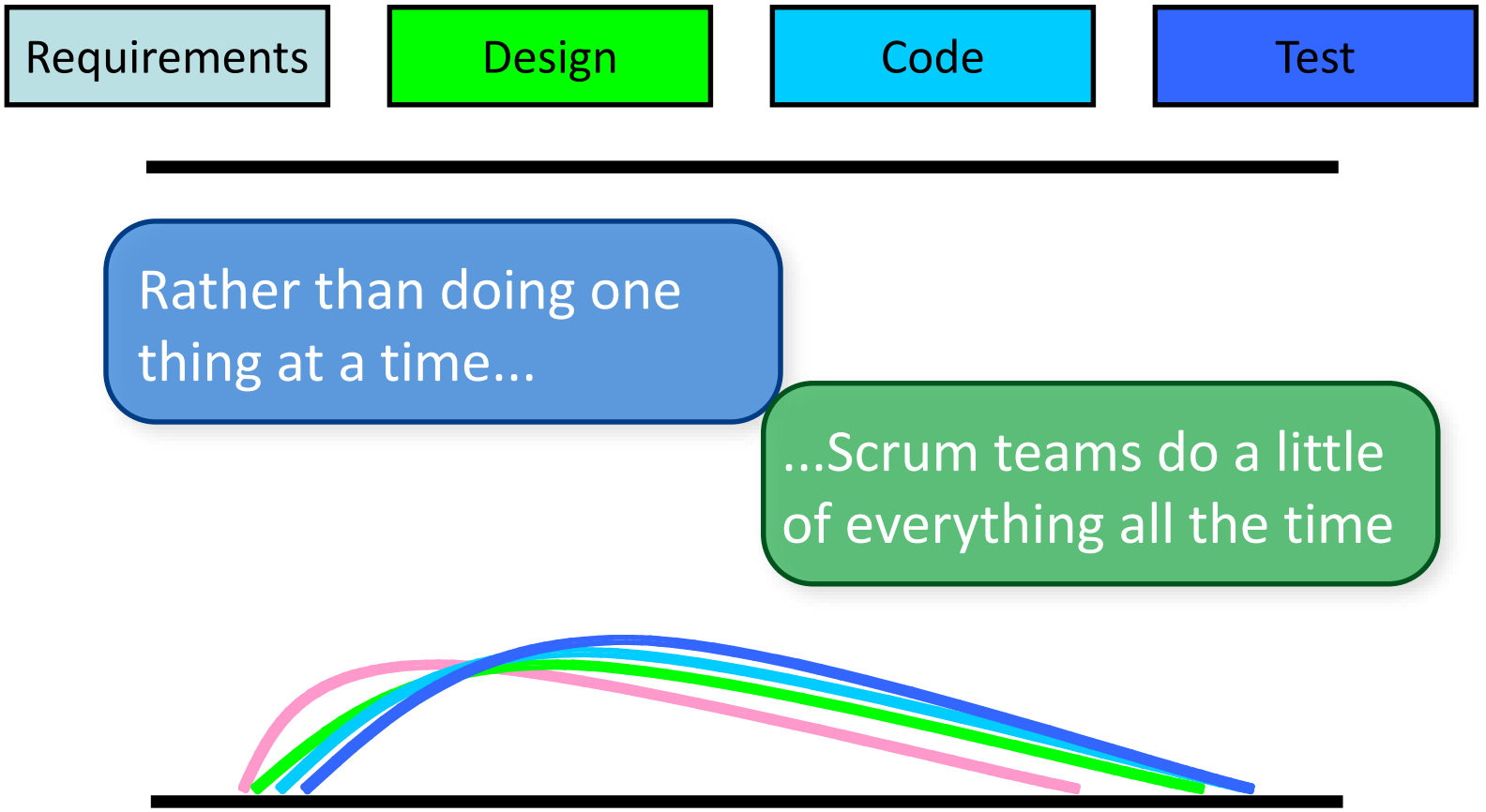
L2.7 – Scrum Key Characteristics

- Self-organising teams
- Product progresses in a series of focused sprints
- Requirements are captured as items in a list of product backlog
- Scrum is one of the agile processes – the one most widely used, discussed and debated
- Time frame is contained to a manageable size (weeks or months)

L2.7 – Scrum Framework

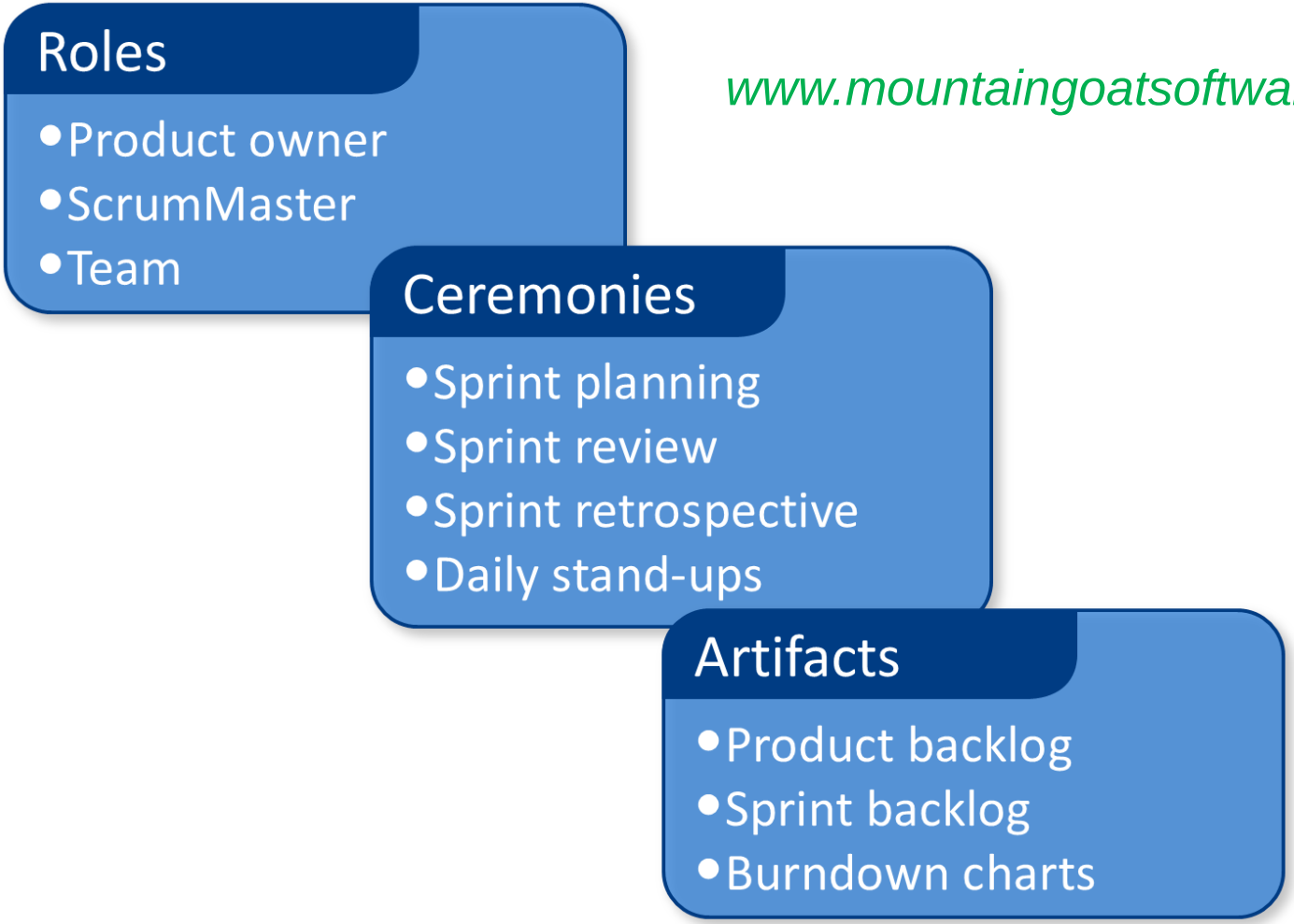


L2.7 – Scrum Framework - *Sprints*



Source: "The New New Product Development Game" by Takeuchi and Nonaka. Harvard Business Review, January 1986.

L2.7 – Scrum Framework



www.mountaingoatsoftware.com

L2.7 – Scrum Roles

Roles

- Product owner
- ScrumMaster
- Team

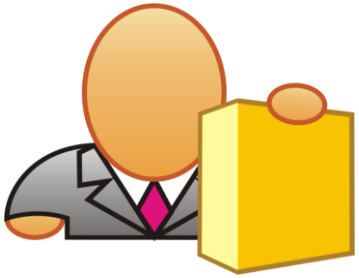
Ceremonies

- Sprint planning
- Sprint review
- Sprint retrospective
- Daily stand-ups

Artifacts

- Product backlog
- Sprint backlog
- Burndown charts

L2.7 – Scrum Roles – *Product Owner*



- Defines the features of the product
- Decides on release date and content
- Is responsible for the Benefits / Profitability of the product (ROI)
- Prioritises features according to market value
- Adjusts features and priority every iteration, as needed
- Accepts or reject work results

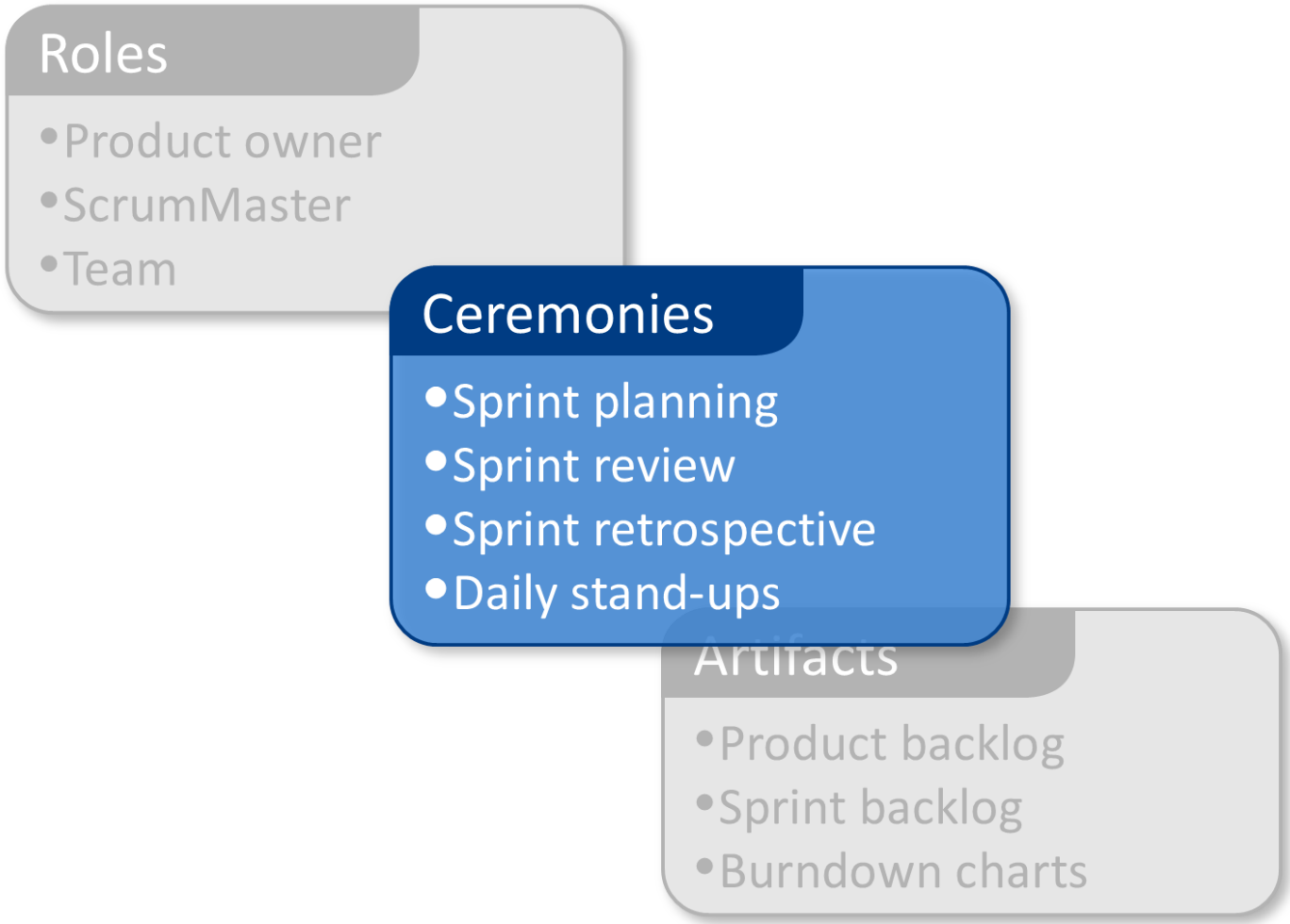
L2.7 – Scrum Roles – *Scrum Master*

- Represents management to the project
- Responsible for enacting Scrum values and practices
- Removes impediments / road blocks
- Ensures that the team is fully functional and productive
- Enables close cooperation across all roles
- Shields the team from external interferences
- Is a member & active participant of the Scrum Team

L2.7 – Scrum Roles – *The Team*

- Typically 5 - 9 people
- Cross-functional:
  - Programmers, testers, user experience designers, business representatives etc.
- Members should be full-time – some exceptions
- Co-located (physically or virtually)

L2.7 – Scrum Ceremonies / Meetings



L2.7 – Scrum Ceremonies / Meetings

Sprint Planning

- Defines how to achieve sprint goal (design)
- Create sprint backlog (User Stories) from product backlog
- Estimate sprint backlog in team velocity and Story Points
- Product Owner priority guides the work
- Release Plan is created
- High-level design is considered

L2.7 – Scrum Ceremonies / Meetings

Sprint Planning

The **Sprint Planning** session is a meeting with the whole team, including the Scrum Master and Product Owner. For a 30-day Sprint it's timeboxed to 8 hours, for 2-week Sprints it's 4 hours, and other Sprint lengths have proportionally sized timeboxes. It's divided into parts, each timeboxed to half of the meeting length:

- In the first half, the team figures out ***what*** can be done in the Sprint. First the team writes down the **Sprint Goal**, a one- or two-sentence statement that says what they'll accomplish in the Sprint. Then they work together to pull items from the Product Backlog to create the **Sprint Backlog**, which has everything they'll build during the Sprint.
- In the second half, they figure out ***how*** the work will get done. They break down (or **decompose**) each item on the Sprint Backlog into **tasks** that will take one day or less. This is how they create a ***plan*** for the Sprint.

Source: Head First Agile – A Brain-Friendly Guide to Agile Principles, Ideas, and Real-World Practices By Andrew Stellman, Jennifer Greene

L2.7 – Scrum Ceremonies / Meetings

Daily Stand-up

- Parameters
  - Daily
  - 15-minutes
  - Stand-up
- Not for problem solving / Not a status meeting
  - Whole world is invited
  - Only team members, ScrumMaster, product owner, can talk
- Helps avoid other unnecessary meetings
- 3 key questions asked:
  1. What did I do yesterday.
  2. What will I do today.
  3. What is in my way to get my work completed.

L2.7 – Scrum Ceremonies / Meetings

Sprint Reviews - Showcase

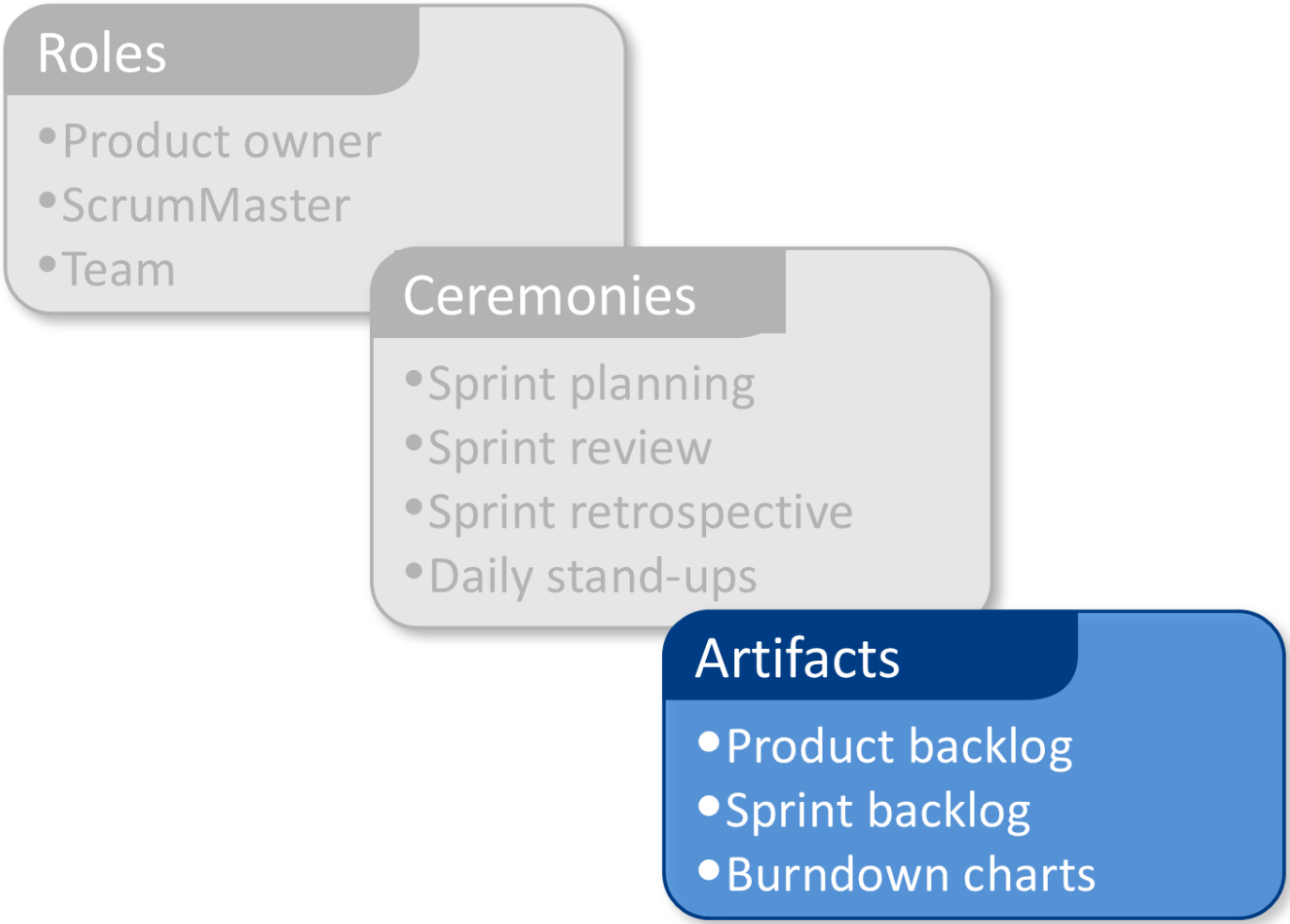
- Team presents what it accomplished during the sprint
- Typically takes the form of a demo of new features or underlying architecture
- Informal
- 2-hour prep time rule
- No slides
- Whole team participates
- Invite the world

L2.7 – Scrum Ceremonies / Meetings

Sprint Retrospective

- Periodically look at what is and isn't working
- Typically 30 minutes
- Done after every sprint
- Whole team participates:
  - ScrumMaster and Team
- Possibly Product Owner, customers and others
- Discuss what to:
  - Start Doing, Stop Doing and Continue Doing

L2.7 – Scrum Artefacts



L2.7 – Scrum Artefacts – Product Backlog

User Stories

- A User Story is a requirement expressed from the perspective of an end-user / customer of the system
- User stories shift the focus from writing about requirements to talking about them
- User stories are short, simple descriptions of a feature told from the perspective of the customer who wants the new capability of the system. They follow a simple template:
  - *As a < type of user >, I want < some goal > so that < some reason >*

L2.7 – Scrum Artefacts – Product Backlog

User Stories

- User stories are written at varying levels of detail.
- They can cover a large amounts of functionality such as this example from a desktop backup product:
  - *As a site visitor, I need to access all news on line*
- Because this level of detail is too large for an agile team to complete in one iteration, it is sometimes split into smaller user stories before it is worked on

L2.7 – Scrum Artefacts – Product Backlog

Story Points

- Story points are a unit of measure for expressing an estimate of the overall effort that will be required to fully implement a product backlog item or any other piece of work
- Story points help estimate how much work can be done in a sprint
- When estimating with story points, a value is assigned to each item. The raw values are unimportant, what matters are the relative values
- A story that is assigned a 2 should be twice as much as a story that is assigned a 1. It should also be two-thirds that is estimated as a 3 story point.
- Instead of assigning 1, 2 and 3, that team could assign 100, 200 and 300. Or 1 million, 2 million and 3 million. It is the ratios that matter, not the actual numbers

L2.7 – Scrum Artefacts – Product Backlog

Product Backlog

- User Story 1
- User Story 2
- User Story 3
- User Story 4
- User Story 5
- User Story nn

- The requirements
- A list of all desired work on the project
- Ideally expressed such that each item has value to the users or customers of the product
- Product Backlog Feature level User Stories are selected for a Sprint by Product Owner
- Reprioritised at the start of each sprint

L2.7 – Scrum Artefacts – Product Backlog Example - Professional Body Website

Product Backlog

- **User Story 1**
- **User Story 2**
- **User Story 3**
- **User Story 4**
- **User Story 5**
- **User Story *nn***

News Section

- User Story 1 - As a site visitor, I can read current news on the home page
- User Story 2 - As a site visitor, I can email news items to the editor
- User Story 3 - As a site member, I can subscribe to an RSS feed of news
- User Story 4 - As a site visitor, I can access old news that is no longer on the home page

Courses and Events

- User Story 5 - As a site visitor, I can see a list of all upcoming “Certification Courses.” I can page through them if there are a lot
- User Story 6 - As a site visitor, I can see a list of all upcoming “Other Courses” (non-certification courses). I can page through them if necessary
- User Story 7 - As a site visitor, I can see a list of all upcoming “Events.” (Events are things such as the conferences, free seminars, etc.)

www.mountaingoatsoftware.com/agile/scrum/scrum-tools/product-backlog/example

L2.7 – Scrum Artefacts – Product Backlog Example - Professional Body Website

Product Backlog

- **User Story 1**
- **User Story 2**
- **User Story 3**
- **User Story 4**
- **User Story 5**
- ***User Story nn***

News Section **- Total of 10 Story Points**

- User Story 1 - As a site visitor, I can read current news on the home page – *1 Story Points*
- User Story 2 - As a site visitor, I can email news items to the editor – *2 Story Points*
- User Story 3 - As a site member, I can subscribe to an RSS feed of news – *3 Story Points*
- User Story 4 - As a site visitor, I can access old news that is no longer on the home page – *4 Story Points*

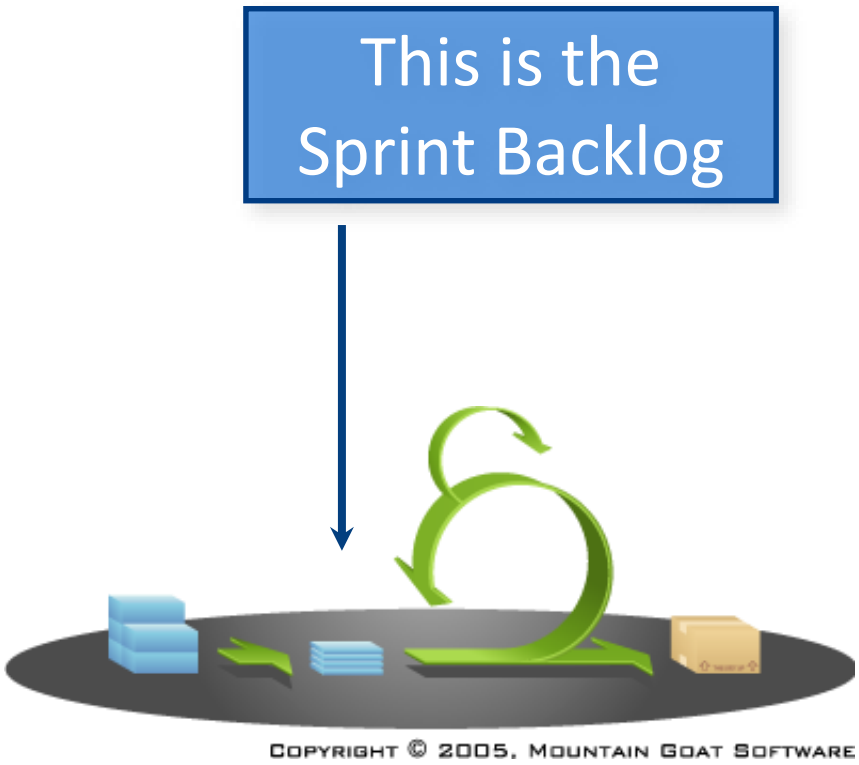
Courses and Events **- Total of 8 Story Points**

- User Story 5 - As a site visitor, I can see a list of all upcoming "Certification Courses." I can page through them if there are a lot – *2 Story Points*
- User Story 6 - As a site visitor, I can see a list of all upcoming "Other Courses" (non-certification courses). I can page through them if necessary – *5 Story Points*
- User Story 7 - As a site visitor, I can see a list of all upcoming "Events." (Events are things such as the conferences, free seminars, etc.) – *1 Story Points*

To complete this total Product Backlog would take 18 Story Points

www.mountaingoatsoftware.com/agile/scrum/scrum-tools/product-backlog/example

L2.7 – Scrum Artefacts – Sprint Backlog / User Story

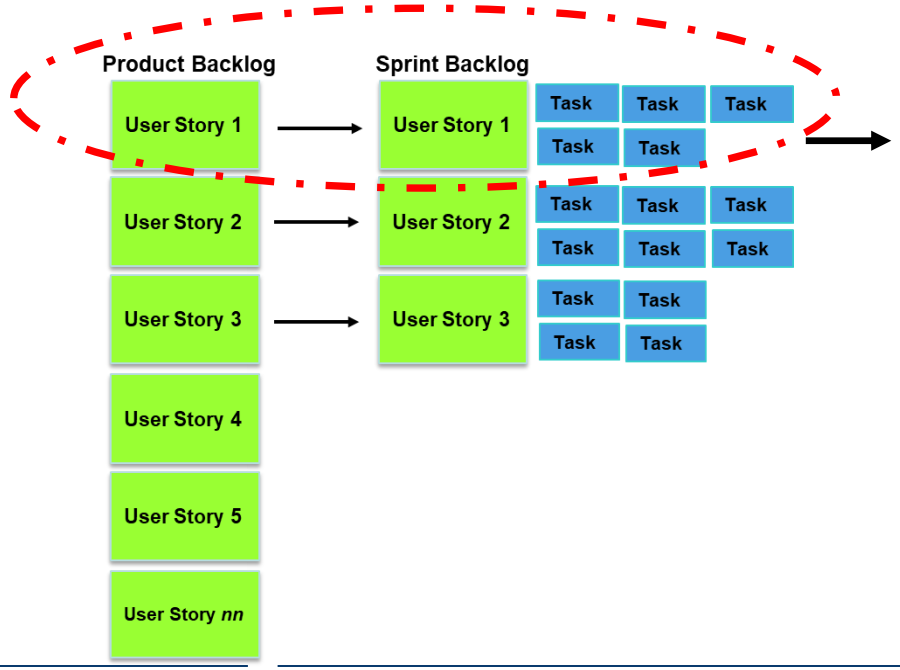


- Scrum team decompose User Stories to a Low level User Stories during Sprint Planning
- The User Stories are used for a conversation between the SME and developer. Developer updates the User Stories with the tasks and hours estimates, "Just-In-Time“
- Remaining estimated items are updated daily
- Sprint Backlog is seldom altered
- User stories in the sprint are either completed 100% or not done

L2.7 – Scrum Artefacts – Sprint Backlog / User Story Example - Professional Body Website

***Sprint Backlog - News Section* – 6 Story Points**

- User Story 1 - As a site visitor, I can read current news on the home page – *1 Story Points*
- User Story 2 - As a site visitor, I can email news items to the editor – *2 Story Points*
- User Story 3 - As a site member, I can subscribe to an RSS feed of news – *3 Story Points*



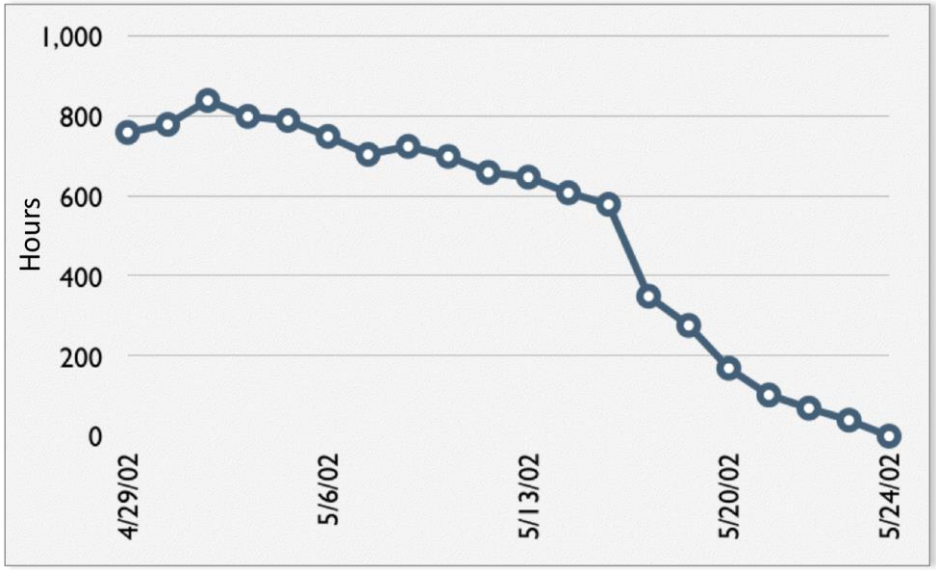
User Story 1 – 1 Story Point – Task Breakdown

Tasks	Mon	Tues	Wed	Thurs	Fri
Code the user interface	8	4	8		
Code the middle tier	16	12	10	4	
Test the middle tier	8	16	16	11	8
Write online help	12				
Write the foo class	8	8	8	8	8
Add error logging			8	4	

Tasks are broken down in hours and managed via a Burn Down chart which allows the team to manage each activity closely

L2.7 – Scrum Artefacts – Burn Down Chart

- A burn down chart is a graphical representation of work left to do versus time.
- The outstanding work (or backlog of user stories) is often on the vertical axis, with time along the horizontal.
- It is used to predict when all of the work will be completed.



L2.7 – Scrum Summary

- Lets have another look at ***SCRUM***

https://www.youtube.com/watch?v=9TycLR0TqFA

L2 - Intended Learning Objectives

5. ~~Understand what Agile is and its origins.~~
6. ~~Understand the Agile framework.~~
7. ~~Understand Scrum – Roles, Ceremonies and Artefacts.~~
8. Understand advantages / disadvantages of Agile.
9. Understand key questions that will help select which approach to use and some examples.

L2.8 – Agile – Advantages & Disadvantages

Advantages

- Customer satisfaction by rapid, continuous delivery of usable software
- People and interactions are emphasised rather than process and tools
- Continuous attention to technical excellence, good design and quality
- Regular adaptation to changing circumstances

Disadvantages

- Difficult to assess the effort required at the beginning
- Can be very demanding (from traditional approaches) on users time
- Harder for new starters to integrate into the team
- Agile is a very different approach – It can be intense for the team
- Requires experienced resources (which are limited in today's market)

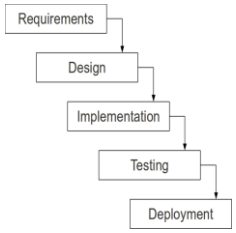
L2 - Intended Learning Objectives

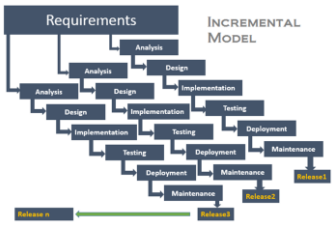
5. ~~Understand what Agile is and its origins.~~
6. ~~Understand the Agile framework.~~
7. ~~Understand Scrum – Roles, Ceremonies and Artefacts.~~
8. ~~Understand advantages / disadvantages of Agile~~.
9. Understand key questions that will help select which approach to use and some examples.

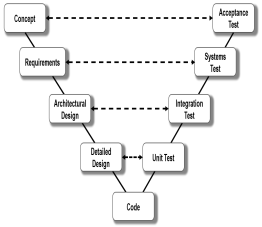
L2.9 – Formal or Agile which one Should I use???

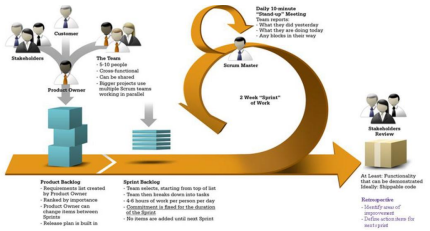
There is no one right answer. The following questions can assist deciding:

- How Stable Are the Requirements?
- Do the end users need to collaborate?
- Is the Time Line Aggressive or Conservative
- What Is the Size of the Project
- Where Are the Project Teams Located
- What Are the Critical Resources?









Agile Where to find out more information

- *www.agilealliance.org*
- *www.mountaingoatsoftware.com/scrum*
- *www.scrumalliance.org*
- *www.controlchaos.com*

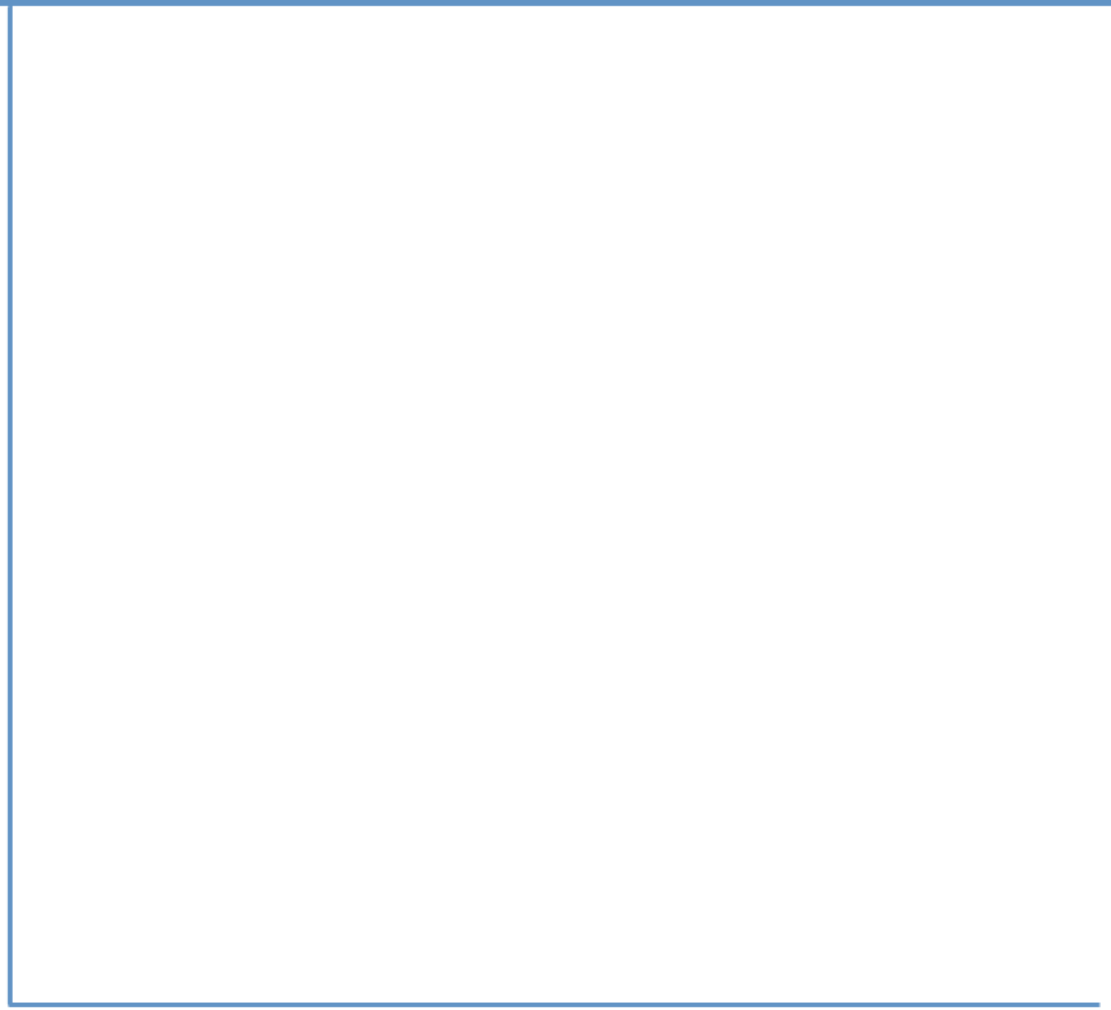
L2 – Quiz

1. Feed back from previous semester.
2. Not marked.
3. Exam multiple choice questions will be selected from these.

Text **HARRYDRAKOS482** to **+61 427 541 357** once to join,

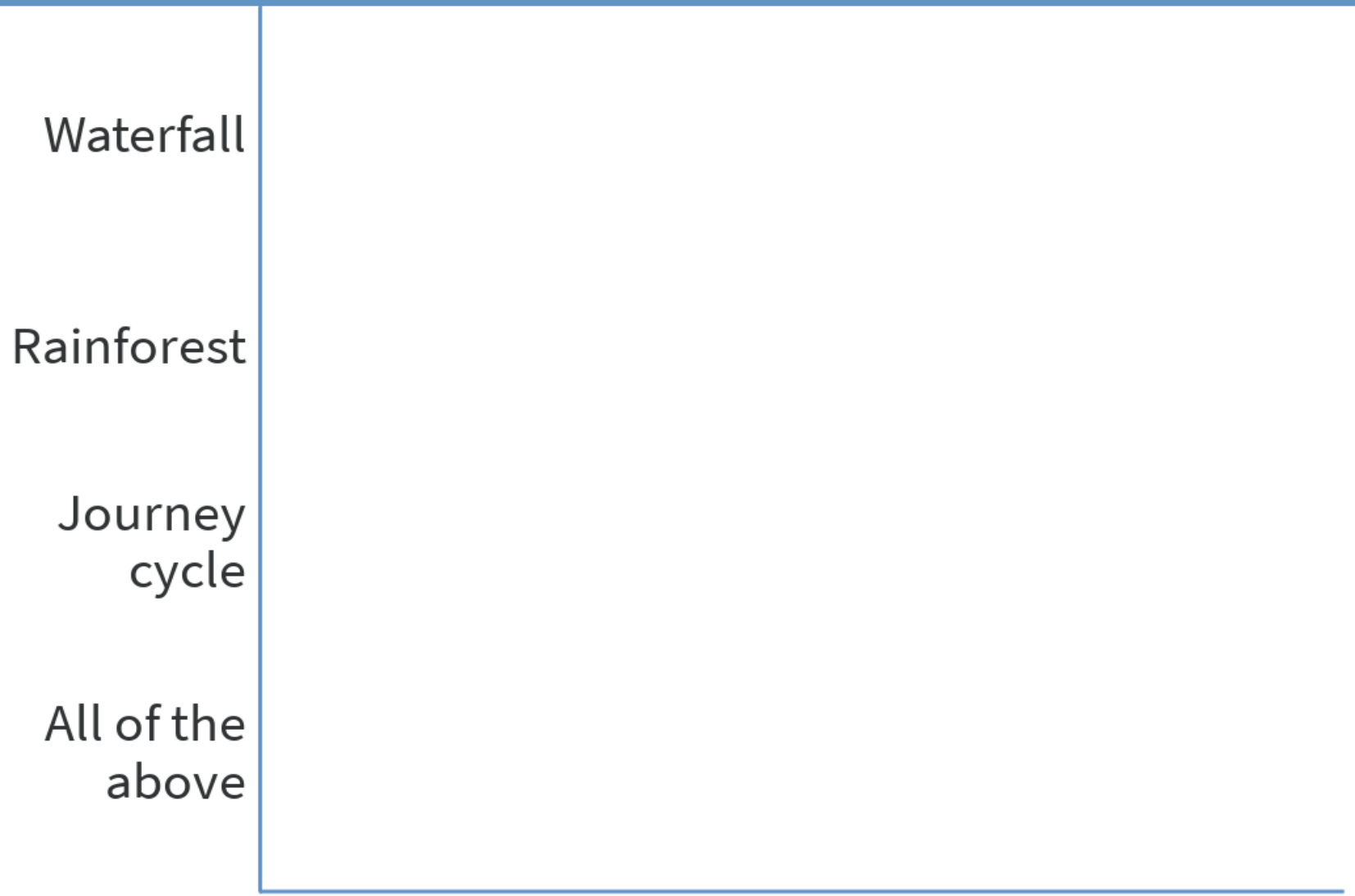
Which of the following information is typically captured in a Project Management Plan?

- Executive Summary
- Project costs and benefits / value
- Project milestones
- Project team / resources
- All of the above



Start the presentation to see live content. Still no live content? Install the app or get help at **PollEv.com/app**

Which of the following is a formal SDLC?



What is Agile?

A set of methodologies based on iterative development where requirements and solutions evolve through collaboration between self-organising cross-functional teams

A software development methodology that does not require any structure or documentation

An approach that enables developers and projects to self-select and deliver what they feel is the best solution for the customer

The way you describe the flexibility of the development / project team working on specific activities

What are the key characteristics of Scrum?

A process that allows teams to focus on delivering the highest business value in the shortest time

The business sets the priorities. Our teams self-manage to determine the best way to deliver the highest priority features

Continued delivery of working software - every two weeks to a month anyone working

All of the above

In which Scrum ceremony does the team provide an update and review activities?

Sprint Planning	**A**
Sprint Initialisation	**B**
Sprint Review	**C**
Daily Stand-ups	**D**
Sprint Retrospective	**E**